



Java is a trademark of Sun Microsystems, Inc.

# JavaOne<sup>SM</sup>

## Swing Filthy-Rich Clients on Mobile Devices with LWUIT

Shai Almog  
Chen Fishbein  
Sun Microsystems

# Agenda

- What is LWUIT?
- LWUIT Basics
- Demo Preface
- LWUIT Chat Client in a few steps
- Community Update
- Thank You!
- Q&A

# What is LWUIT?

- Advanced UI library
- Compelling UI
- Consistent across devices
- Mass market devices - MIDP 2.0, CLDC 1.1
- Portable
  - MIDP, CDC, SE, BB, Android...
- Inspired by Swing
- Easy deployment - is it even possible in Java™ ME?



# LWUIT Basics

- Lightweight UI Toolkit - Lightweight means we can "tinker"
  - Everything in LWUIT is painted in Java
  - We can change **EVERYTHING**
- We can leverage device Media, SVG & 3D
- Customizing is easy - take advantage of that







**LWUIT**

myInstinct Bible

Comverse

Mobits

CTIA conference

WeightWatch

myInstinct Bible

Cycloscope

MajiPic

Telmap

rSyncFM

Genaker

Emoze

Nuux

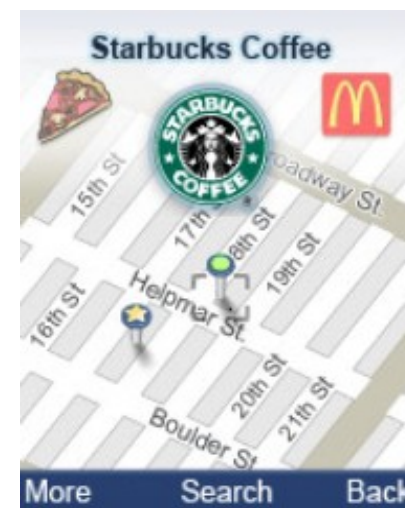
MajiPlayer

Posterita

FootyMad mobile

## Demo Preface

- We are developers not designers
- For us "pretty" == "useful", effects must have a meaning:
  - Colors should highlight & define
  - Animations should indicate something
- Be innovative - take advantage of the API



# Chat Client

- Simple demo application
- We made it typically plain
- We will focus both on aesthetics of fixing this and on its look & feel



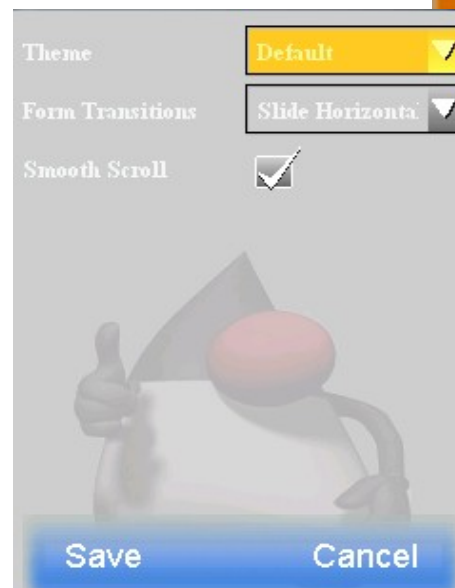
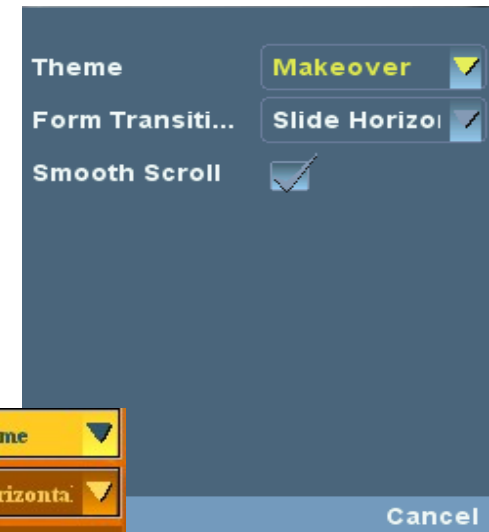
# Try The New Chat Client

- If actions speak louder than words then an interactive demo is a scream
- Pull out your mobile phone and start chatting with us!
- Please be polite & respectful...
- <http://majimob.com/ChatServer>



# Theme It

- Theming makes all the difference
- Cost for developers == 0 lines of code



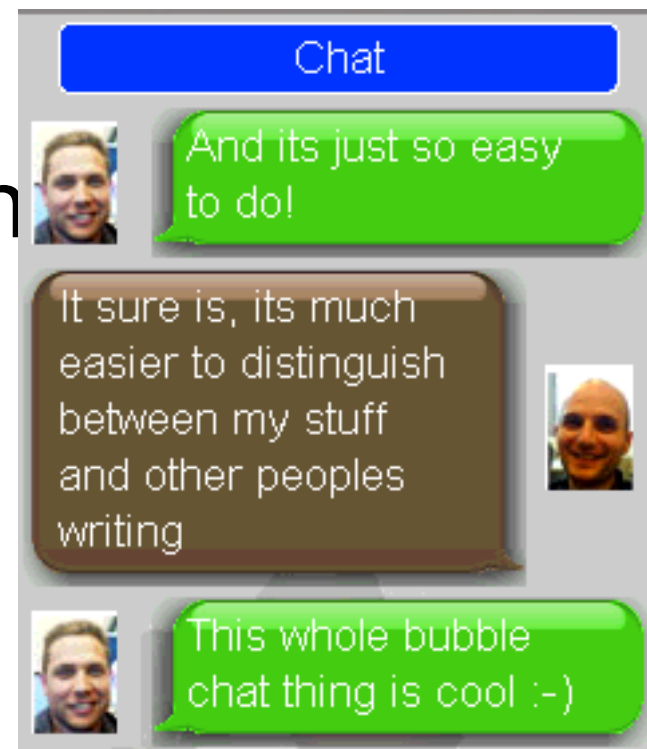
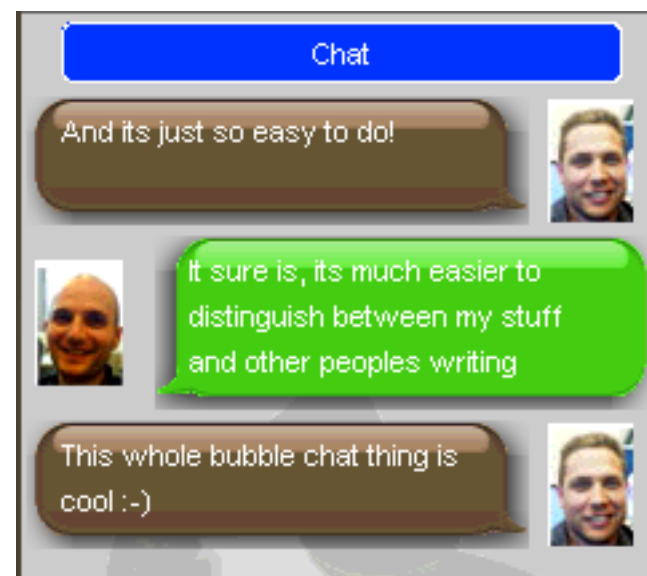
# Alignment

- Entries in the chat don't align properly...
- Problem: Icon/text size can differ between entries
- Use `setPreferredW`, Loop over elements and align with an upper limit
- Our sense of aesthetics is influenced heavily by alignment



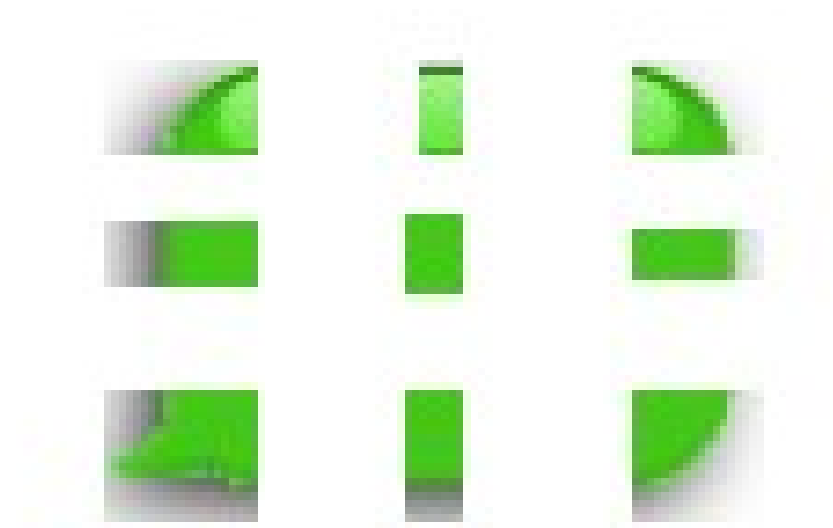
# Bubbles

- Bubble chat allows differentiation in a glance of me vs. them
- We reverse layout in a single point to invert sides
- Applying styles to existing components can be tricky with themeing
- Bubbles are borders - more efficient, no scaling artifacts



## 9 Part Border

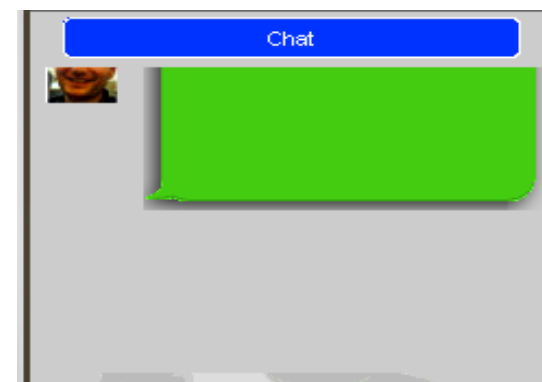
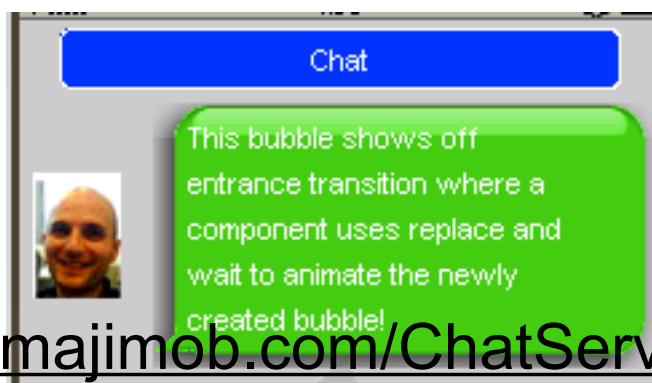
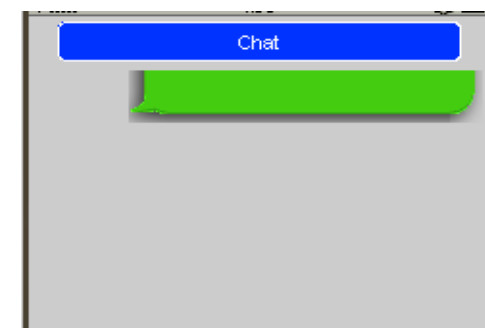
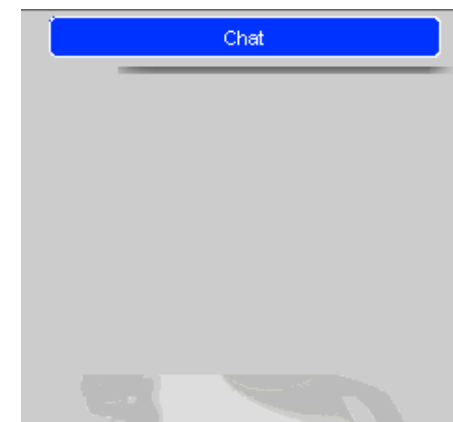
- Bubbles use the 9 image border style supported by LWUIT
- We can assemble such a border using the LWUIT Designer



# Animate Entrance

- Bubbles enter the screen with an animation transition
- Uses `replace()` on a dummy component to make room

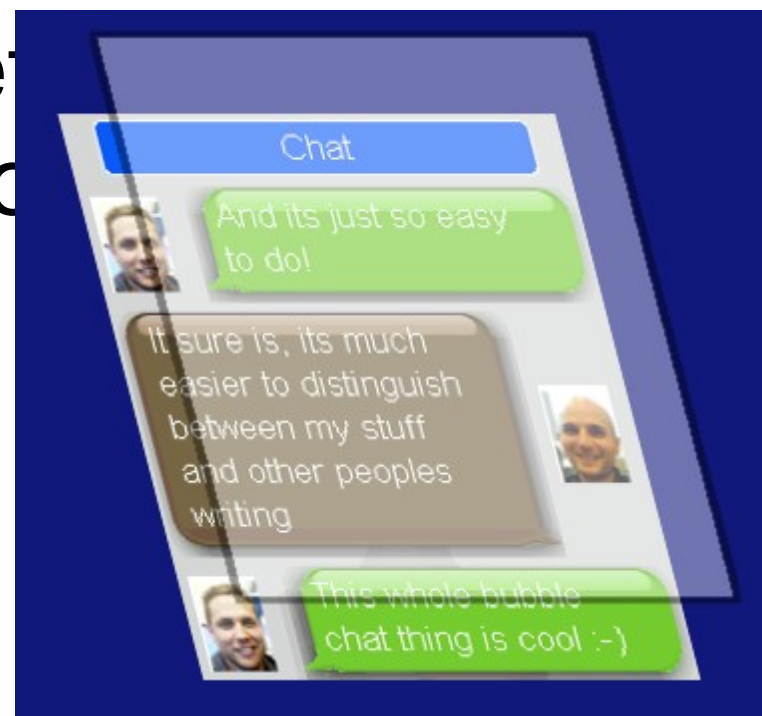
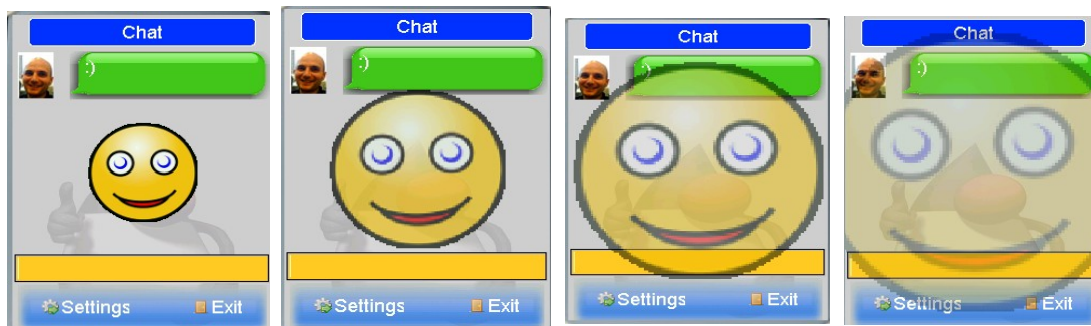
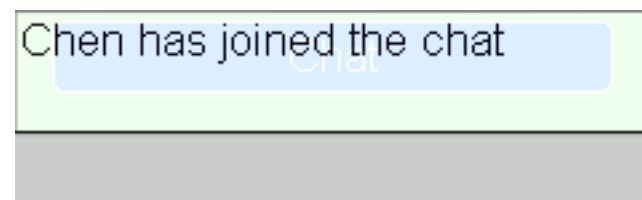
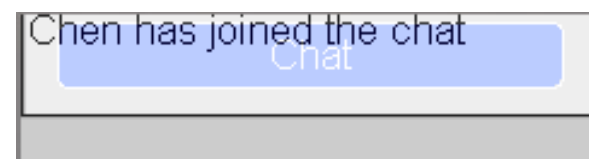
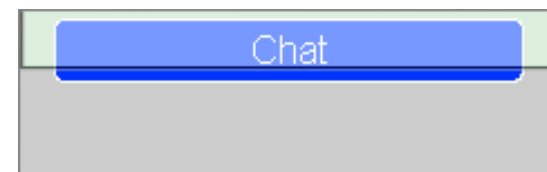
```
dummy.setPreferredSize(e.getPreferredSize());  
chatScreen.addComponent(0, dummy);  
e.requestFocus(); revalidate();  
chatScreen.replaceAndWait(dummy, e, slide);
```





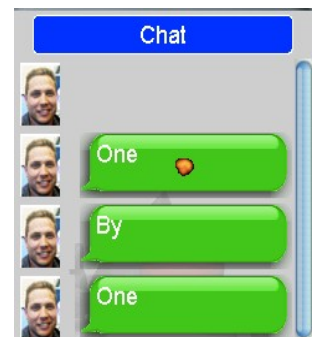
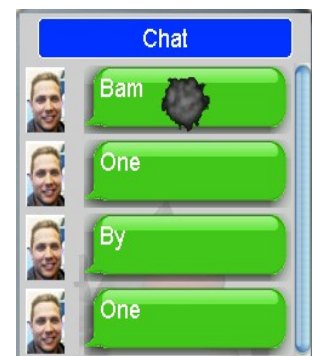
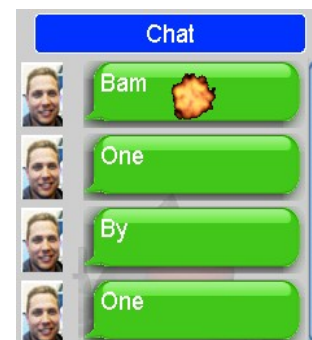
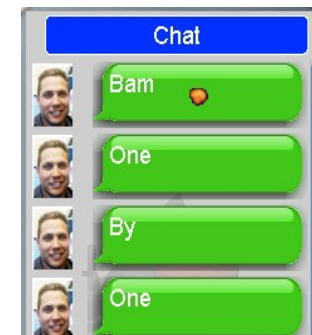
# Glass Pane Effects

- Glass pane is great for global animations
- For waiting use Dialogs (they can be transparent)
- GlassPanels are great for elegant and non intrusive information delivery



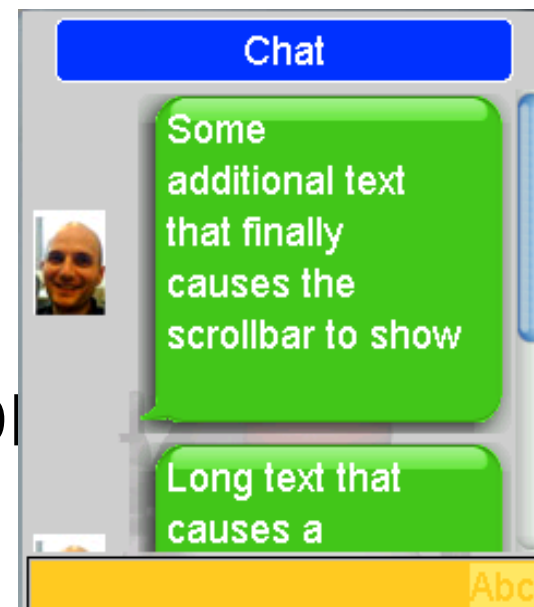
# Transitions - Make Your Own

- We over emphasized the 3D transitions
- A transition can be ANYTHING!
  - Exploding components
  - Component "motion" transition as part of the demo
  - Screen rotation
- Make your own, its easy...



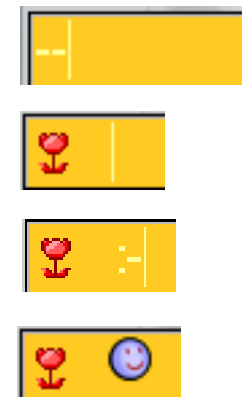
# Customized Look & Feel

- Look & Feel is the central point where everything is "drawn"
- We can customize the application globally
- Modify almost every aspect of LWUIT from scrollbars to checkboxes
- Even allow for effects such as fading scrollbars



# A Rose Is Not a Rose ;-) --@ LOL

- Typing :) into a text field should produce an emoticon
- This should appear globally in the application
- We can replace the implementation with a minor tweak to accomplish this & this: --@
- Replace the LWUIT Implementation drawString



## Demo

- We will take down the server after JavaOne to avoid abuse
- Thanks for the guys at majimob for hosting it, you can try it live here (go with your cell phone to): <http://majimob.com/ChatServer/>
- We will try to make the source for both the demo and the server will be available at the LWUIT or LWUIT Incubator project
- Demo is 298kb, 144kb without resources



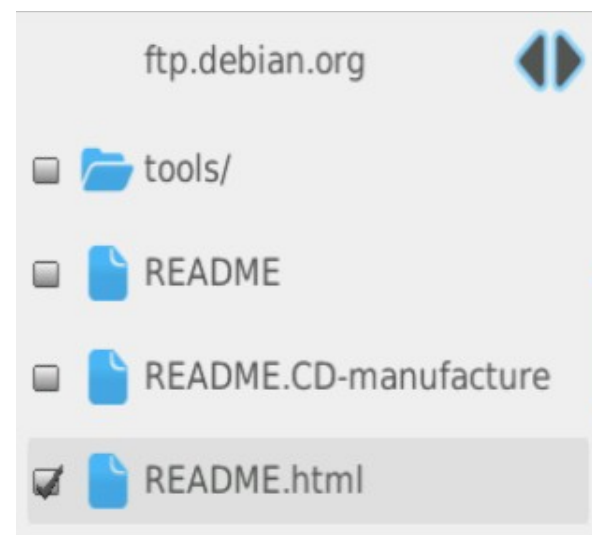
# Community Update

- Top 5 project list by email traffic
- 85,000+ views/8,000+ messages
- 60+ Observers
- 8 3rd party developers signed SCA, Thorsten created the BB & Android ports
- 2 books in progress
- Being adopted in the DTV open spec (Ginga-J)



# Thank You!

- LWUIT wouldn't be anything without the community
- Your contribution, support and advocacy has been wonderful and helpful!
- Thanks to Thorsten for a great contribution



## Q&A



## Find Out More

- BoF: Q&A BoF-4551 at 20:30 here
- <http://lwuit.dev.java.net/>
- <http://lwuit.blogspot.com/>
- Place your application in the featured apps page:  
<https://lwuit.dev.java.net/nonav/featuredapps>
- LWUIT Support forum:  
<http://forums.java.net/jive/forum.jspa?forum>



# JavaOne<sup>SM</sup>

# Thank You

Shai Almog & Chen  
Fishbein

shai.almog@sun.com,  
chen.fishbein@sun.com





# Alignment Source Sample

```
Component alignedCmp = (Component)aligned.elementAt(0);
int alignedW = alignedCmp.getPreferredW();
aligned.addElement(c);
if(alignedW >= w) {
    c.setPreferredW(alignedW);
    return;
}
if(alignedW < w) {
    for(int iter = 0 ; iter < aligned.size() ; iter++) {
        ((Component)aligned.elementAt(iter)).setPreferredW(w);
    }
    return;
}
```

# Bubbles Source Sample

```
public Text(String t, boolean me) {  
    super(t, 1, 20);  
    this.me = me;  
    // the super class constructor initializes the style before  
    // "me" has a chance to initialize the local variable  
    Style s =  
        UIManager.getInstance().getComponentStyle(getUIID());  
    setSelectedStyle(s); setUnSelectedStyle(s);  
}  
public String getUIID() {  
    if(me) return "TextAreaMe";  
    return "TextAreaThem";  
}
```